

# Aplikasi Pendeteksi Dan Analisa Cuaca Menggunakan Metode K-Nearest Neighbor Berbasis Android

Jaguari Brawijaya<sup>1,\*</sup>, Felix<sup>2</sup>, Kristien Margi Suryaningrum<sup>3</sup>

Universitas Bunda Mulia, 4, Kota Tua, Jl. Lodan Dalam No.1, RT.9/RW.8, Ancol, Kec. Pademangan, Kota Jkt Utara, Daerah Khusus Ibukota Jakarta 14430, Indonesia

<sup>1</sup> brawijayajag@gmail.com\*; <sup>2</sup> kobayakawaringo@gmail.com, <sup>3</sup> kristienmargi@gmail.com

\* Penulis Korespondensi

## Abstrak

Cuaca adalah faktor penentu dalam kehidupan sehari-hari makhluk hidup, terutama manusia, cuaca sendiri adalah sebuah fenomena alam yang terjadi di atmosfer Bumi dan planet lainnya. Cuaca adalah suatu aktivitas yang dapat berlangsung selama beberapa hari. Cuaca dengan aktivitas jangka waktu yang lebih lama lebih dikenal sebagai iklim. Cuaca terjadi karena reaksi suhu dan kelembaban yang berbeda antara satu tempat dengan tempat lainnya. Perbedaan ini bisa terjadi karena sudut pemanasan matahari yang berbeda dari satu tempat ke tempat lainnya.

Kata Kunci: *Cuaca, K-Nearest Neighbor, Android, Pendeteksi Cuaca, Pendeteksi Citra, Nilai Warna, RGB*

## 1. Pendahuluan

Pada zaman sekarang yang terpengaruh oleh pemanasan global, cuaca dapat berubah secara pesat, sehingga ramalan cuaca dapat berubah sewaktu-waktu dan menyebabkan aktivitas terganggu bila turun hujan secara tiba-tiba. Meramal cuaca dengan mengidentifikasi jenis awan yang berada di langit sudah dilakukan manusia sejak dahulu namun sudah terlupakan dengan lebih mengandalkan ramalan cuaca di Internet.

Walaupun Internet menyediakan banyak ramalan cuaca yang terpercaya tidak semua orang mempunyai akses Internet yang menyebabkan mereka terjebak cuaca yang tidak mereka inginkan ketika berada di luar, maka dari itu sebagai suatu upaya untuk mencegah masalah ini, pembuatan sebuah sistem yang dapat mendeteksi jenis dan dapat mengidentifikasi ciri-ciri awan.

## 2. Landasan Teori

### 2.1. Cuaca

Cuaca merupakan suatu kondisi udara di suatu tempat pada waktu yang relatif singkat, yang dinyatakan dengan nilai beberapa parameter seperti suhu, tekanan udara, kecepatan angin, kelembaban udara, dan berbagai fenomena atmosfer lainnya.

### 2.2. Basis Data

Database adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut (Alhadi Saputra, 2013). Sebuah Database Management System (DBMS) merupakan suatu kumpulan data yang saling berhubungan dan merupakan satu set program untuk mengakses data tersebut. Kumpulan data yang biasa disebut database berisi informasi yang relevan untuk suatu perusahaan. Tujuan utama dari DBMS adalah untuk menyediakan cara untuk menyimpan dan mengambil informasi database yang baik, nyaman, dan efisien.

### 2.3. Flowchart

Flowchart atau Bagan alir merupakan teknik analitis yang digunakan untuk menjelaskan aspek-aspek sistem informasi secara jelas, tepat dan logis. Bagan alir menggunakan serangkaian simbol standar untuk menguraikan prosedur pengolahan transaksi yang digunakan oleh sebuah perusahaan, sekaligus menguraikan aliran data dalam sebuah sistem.

### 2.4. DFD (Data Flow Diagram)

DFD adalah Suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan ke mana tujuan data yang keluar dari sistem, di mana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan.

### 2.5. K-Nearest Neighbor

K-Nearest Neighbor (KNN) merupakan metode yang biasa digunakan pada klasifikasi data. Algoritma ini digunakan untuk mengklasifikasikan terhadap objek berdasarkan data pembelajaran yang jarak tetangganya paling dekat atau memiliki nilai selisih yang kecil dengan objek tersebut.

## 3. Metode K-Nearest Neighbor

KNN merupakan suatu metode yang menggunakan algoritma *supervised* dengan hasil dari *query instance* yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada KNN. Tujuan dari algoritma ini ialah mengklasifikasikan objek baru berdasarkan atribut dan *training sample*. Prinsip umum dari algoritma ini adalah menemukan *k data training* untuk menentukan *k-nearest neighbor* berdasarkan ukuran jarak. Selanjutnya mayoritas dari *k* tetangga terdekat akan menjadi dasar untuk memutuskan kategori dari *sample* berikutnya. Selain itu algoritma ini sendiri sering digunakan untuk klasifikasi pada teknik *data mining* meskipun dapat digunakan untuk estimasi dan prediksi data.

### 3.1. Analisis Algoritma K-Nearest Neighbor

Dalam proses pengembangan aplikasi ini terdapat pilihan algoritma yang beragam, namun kami memilih algoritma *K-Nearest Neighbor* dikarenakan algoritma ini dapat diimplementasi dengan baik dalam kasus penentuan cuaca, yang di mana jumlah variabel yang dibutuhkan dalam proses perbandingan citra langit tidak besar, dan algoritma K-NN bekerja lebih baik dalam lingkungan data yang tidak memiliki terlalu banyak variabel. Algoritma ini melakukan proses klasifikasi berdasarkan jarak antara data *testing* dengan data-data *training*. Jarak ini disebut dengan *Euclidean distance*.

### 3.2. Proses Penelitian Penggunaan Algoritma K-NN

Berdasarkan hasil analisa terhadap algoritma K-NN, berikut tahap – tahap percobaan dijelaskan pada Gambar 1 yang digunakan untuk mendasari proses cara penggunaan algoritma K-NN:

1. Diberlakukan proses pengumpulan citra langit menggunakan kamera, yang menghasilkan citra digital dengan ekstensi .jpeg (*Joint Photographic Experts Group*)
2. Kemudian citra langit yang dikumpulkan melalui proses *preprocessing*, di mana citra tersebut melalui proses *cropping* yang menghasilkan citra tersebut diperkecil menjadi gambar dengan resolusi 480 x 480p (*pixels*). Lalu citra yang di *preprocessing* dibagi menjadi 2 jenis: *data training*, dan *data testing*.
3. Selanjutnya gambar tersebut diklasifikasikan atas beberapa kondisi cuaca: cerah, berawan, hujan dalam 2 waktu, yakni: pagi/malam.
4. Proses ekstraksi ciri dari citra untuk mendapatkan karakter warna RGB (*Red, Green, Blue*) pada citra langit. Proses ini kemudian dilakukan berulang-ulang kali untuk mencari nilai rata - rata yang berfungsi untuk mendefinisikan *range* warna, yaitu *threshold* pada setiap karakter warna, yang berfungsi sebagai proses *data training*.
5. Dari *data training* yang didapatkan, diberlakukan klasifikasi kondisi cuaca berdasarkan dari nilai rata - rata RGB, yang kemudian digunakan sebagai pembanding kepada *data testing*.

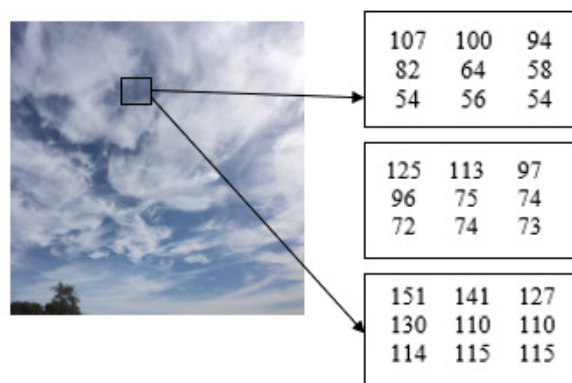


Gambar 1 Flowchart Proses Penelitian

### 3.3. Implementasi Algoritma K-Nearest Neighbor Terhadap Aplikasi

Berdasarkan penelitian tentang cara penggunaan algoritma K-NN, berikut adalah proses implementasi algoritma K-NN ke dalam aplikasi pendeteksi kondisi cuaca:

1. Pengguna meng-*input* citra langit digital ke dalam sistem (dengan menggunakan kamera ataupun dari galeri *handphone*).
2. *Input* citra digital kemudian melalui proses *preprocessing* di mana citra tersebut di perkecil menjadi citra berukuran 480 x 480p.
3. Citra yang telah melalui proses *preprocessing* kemudian melalui proses ekstraksi ciri ditunjukkan pada Gambar 2.



Gambar 2 Contoh cuplikan hasil ekstraksi nilai RGB pada sampel citra langit.

4. Berdasarkan hasil nilai RGB dalam bentuk sebuah matriks 3x3 yang telah diekstraksi diberlakukan perhitungan nilai rata – rata dari nilai RGB ditunjukkan pada Gambar 3.

R

107	100	94
82	64	58
54	56	54

$$r = \frac{R}{(R + G + B)} = \frac{669}{669 + 779 + 1113} = \frac{669}{2561} = 0,261$$

G

125	113	97
96	75	74
72	74	73

$$g = \frac{G}{(R + G + B)} = \frac{779}{669 + 779 + 1113} = \frac{779}{2561} = 0,304$$

B

151	141	127
130	110	110
114	115	115

$$b = \frac{B}{(R + G + B)} = \frac{1113}{669 + 779 + 1113} = \frac{1113}{2561} = 0,434$$

Gambar 3 Contoh perhitungan nilai rata – rata nilai RGB.

- Berdasarkan dari hasil perhitungan nilai rata – rata RGB citra langit yang di-input nilai tersebut di dibandingkan dengan nilai perhitungan rata – rata Data latih dengan menggunakan rumus *Euclidean Distance* ditunjukkan pada Tabel 1 dan Tabel 2.

Tabel 1 Hasil perhitungan rata – rata RGB pada Data Latih

Citra Latih	Ekstraksi Rata – Rata Nilai RGB			Kelas
	R	G	B	
train1.jpeg	0,301	0,310	0,315	Pagi Cerah
train2.jpeg	0,281	0,305	0,440	Pagi Berawan
train3.jpeg	0,256	0,330	0,484	Pagi Hujan
train4.jpeg	0,231	0,352	0,506	Malam Cerah
train5.jpeg	0,240	0,377	0,529	Malam Berawan
train6.jpeg	0,214	0,420	0,551	Malam Hujan

Tabel 2 Hasil perhitungan rata – rata nilai RGB dari data training

Citra Latih	Ekstraksi Rata – Rata Nilai RGB			Kelas
	R	G	B	
train1.jpeg	0,301	0,310	0,315	Pagi Cerah
train2.jpeg	0,281	0,305	0,440	Pagi Berawan
train3.jpeg	0,256	0,330	0,484	Pagi Hujan
train4.jpeg	0,231	0,352	0,506	Malam Cerah
train5.jpeg	0,240	0,377	0,529	Malam Berawan
train6.jpeg	0,214	0,420	0,551	Malam Hujan

Perhitungan *Euclidean Distance* dari citra uji test1.jpeg terhadap citra – citra latih ditunjukkan pada Tabel 3.

Tabel 3 Hasil perhitungan *Euclidean Distance* Data Uji dengan Data Latih

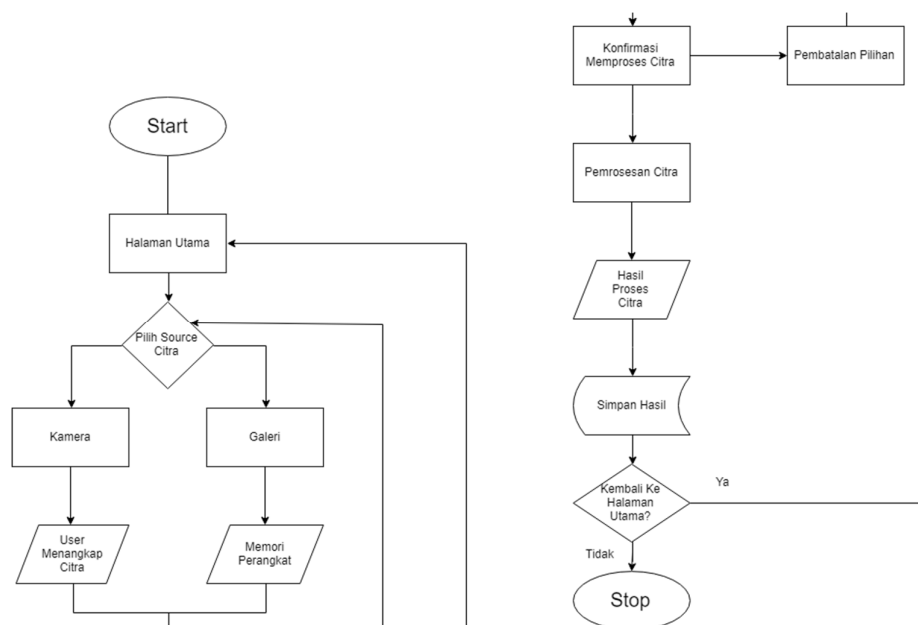
Citra yang Dibandingkan	Perhitungan <i>Euclidean Distance</i>	Hasil
test1 & train1	$\sqrt{(0,301 - 0,261)^2 + (0,310 - 0,304)^2 + (0,315 - 0,434)^2}$	0,015797
test1 & train2	$\sqrt{(0,281 - 0,261)^2 + (0,305 - 0,304)^2 + (0,440 - 0,434)^2}$	0,001637
test1 & train3	$\sqrt{(0,256 - 0,261)^2 + (0,330 - 0,304)^2 + (0,484 - 0,434)^2}$	0,003201
test1 & train4	$\sqrt{(0,231 - 0,261)^2 + (0,352 - 0,304)^2 + (0,506 - 0,434)^2}$	0,008388
test1 & train5	$\sqrt{(0,240 - 0,261)^2 + (0,377 - 0,304)^2 + (0,529 - 0,434)^2}$	0,014795
test1 & train6	$\sqrt{(0,214 - 0,261)^2 + (0,420 - 0,304)^2 + (0,551 - 0,434)^2}$	0,029354

- Berdasarkan hasil perhitungan *Euclidean Distance*, Algoritma K-NN melakukan klasifikasi terhadap data uji. Klasifikasi ini dilakukan dengan cara membandingkan nilai *Euclidean Distance* data uji dengan data – data latih lain. Hasil dari data uji yang melalui klasifikasi K-NN yang didasarkan oleh *Euclidean Distance* ditunjukkan pada Tabel 4.

Tabel 4 Hasil Klasifikasi K-NN berdasarkan nilai *Euclidean Distance*.

Citra Uji	<i>Euclidean Distance</i>	Hasil	Hasil Aktual
test1.jpeg	0,001637	Pagi Berawan	Pagi Berawan
test2.jpeg	0,002817	Pagi Cerah	Pagi Cerah
test3.jpeg	0,003311	Malam Cerah	Malam Cerah
test4.jpeg	0,007372	Malam Berawan	Malam Berawan
test5.jpeg	0,024747	Malam Cerah	Malam Cerah
test6.jpeg	0,031354	Malam Hujan	Malam Berawan

### 3.4. Perancangan Proses



Gambar 4 Flowchart Sistem

Diagram perancangan proses ditunjukkan dengan Gambar 4.

### 3.5. Perancangan Sistem

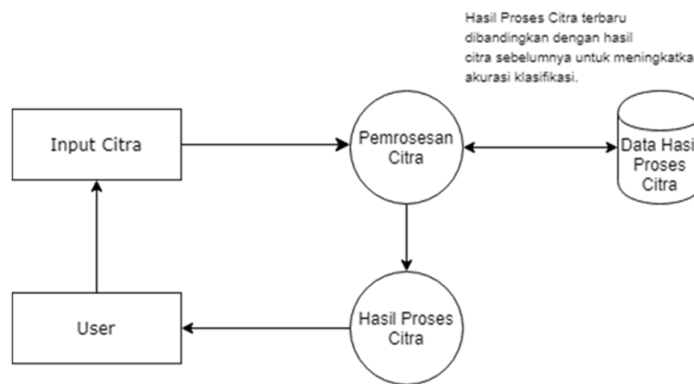
Diagram perancangan sistem DFD Level-0 ditunjukkan dengan Gambar 5.



Gambar 5 DFD Level-0 Context Diagram

### 3.6. DFD Level-1

Pada Gambar 5 dapat diperinci menjadi DFD Level-1. Proses-proses pada level-1 adalah dekomposisi dari proses DFD Level-0 yang ditunjukkan pada Gambar 6.






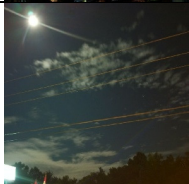
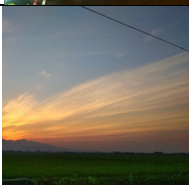

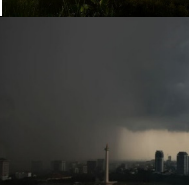



Gambar 6 DFD Level-1 Sistem Pemrosesan Citra

## 4. Pengujian Keberhasilan Algoritma K-NN

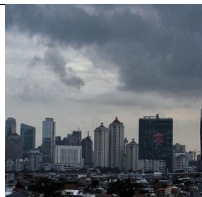
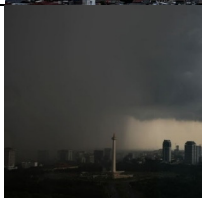

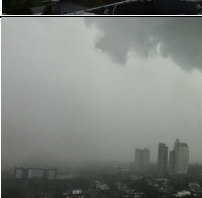

Pengujian keberhasilan metode dalam penelitian ini didasarkan pada hasil prediksi cuaca pada citra digital yang berhasil dikenali. Objek dari pengujian ini adalah citra langit yang diambil menggunakan kamera *smartphone* dan gambar – gambar langit dengan ukuran dan kualitas bervariasi. Pengujian dilakukan dengan menggunakan 100 citra langit dalam kondisi yang bervariasi.

Tabel 5 Hasil pengujian menggunakan citra langit dengan kondisi yang berbeda – beda antara perangkat dengan objek tujuan.

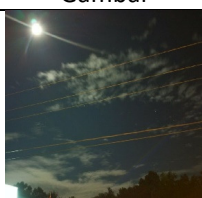
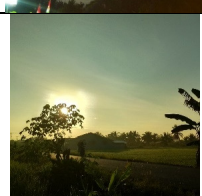
No.	Gambar	R	G	B	Hasil Proses	Hasil Aktual
1		0.27727	0.34091	0.38182	Pagi Cerah	Pagi Cerah
2		0.38182	0.3318	0.33641	Pagi Hujan	Pagi Hujan

3		0.24	0.4	0.36	Pagi Berawan	Pagi Berawan
4		0.66667	0.33333	0.0	Pagi Berawan	Malam Cerah
5		0.46927	0.35196	0.35196	Pagi Berawan	Pagi Cerah
6		0.5	0.5	0.0	Malam Berawan	Pagi Cerah
7		0.39655	0.32759	0.27586	Pagi Berawan	Pagi Berawan
8		0.22222	0.44444	0.33333	Pagi Hujan	Pagi Berawan
9		0.1875	0.4375	0.375	Pagi Hujan	Pagi Hujan
10		0.33333	0.33333	0.33333	Malam Cerah	Malam Cerah


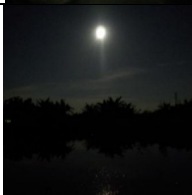

Tabel 6 Hasil pengujian menggunakan citra langit dengan kondisi banyak bangunan dalam citra.

No.	Gambar	R	G	B	Hasil Proses	Hasil Aktual
1		0.25	0.75	0.0	Pagi Berawan	Pagi Hujan
2		0.39655	0.32759	0.27586	Pagi Berawan	Pagi Berawan
3		0.32937	0.3254	0.34524	Pagi Berawan	Pagi Berawan
4		0.34375	0.4375	0.21875	Pagi Hujan	Malam Cerah
5		0.33333	0.33333	0.33333	Malam Cerah	Pagi Cerah

Tabel 7 Hasil pengujian menggunakan citra langit dengan kondisi cahaya lampu/matahari/bulan tertangkap citra.

No.	Gambar	R	G	B	Hasil Proses	Hasil Aktual
1		0.66667	0.33333	0.0	Pagi Berawan	Malam Cerah
2		0.5	0.5	0.0	Malam Berawan	Pagi Cerah



3		0.33333	0.33333	0.33333	Malam Cerah	Malam Berawan
4		0.33333	0.33333	0.33333	Malam Cerah	Malam Cerah
5		0.4	0.21818	0.38182	Malam Berawan	Malam Berawan

Dilakukan pengujian terhadap hasil ekstraksi citra secara acak tanpa memedulikan objek bangunan maupun cahaya yang tertangkap dengan kamera perangkat pengguna serta dengan kondisi citra buram dan didapati hasil 64%. Berdasarkan jurnal [5], kekurangan metode ekstraksi fitur *K-Nearest Neighbor* adalah sensitif terhadap obstruksi *value* yang bukan *value* cuaca. Maka dari itu, untuk membuktikan apakah obstruksi bangunan dan cahaya ikut berpengaruh besar dalam ekstraksi *value* citra, kami melakukan uji coba terhadap citra yang terdapat obstruksi bangunan dan cahaya. Adapun hasil dari uji coba tersebut adalah keberhasilan pengenalan citra dengan obstruksi bangunan sebesar 43,333% dan obstruksi cahaya sebesar 30%. Adapun kesalahan pengenalan citra dalam uji coba obstruksi bangunan dan cahaya ini dapat disebabkan oleh banyaknya bangunan dan cahaya yang meng-obstruksi cuaca terlalu besar, sehingga menyebabkan hasil ekstraksi *value* warna cuaca tercampur dengan *value* warna bangunan dan cahaya. Namun untuk citra yang tidak mengalami obstruksi bangunan dan cahaya, masih dapat dikenali aplikasi dengan baik.

## 5. Kesimpulan

Dari penelitian dan pengujian yang dilakukan maka dapat disimpulkan sebagai berikut:

1. Aplikasi Pendeteksi dan Analisa Cuaca ini dapat membantu masyarakat dan peneliti dalam mengetahui nilai rata-rata warna dari suatu cuaca dan sebagai sumber informasi cuaca pada saat aplikasi digunakan.
2. Aplikasi menggunakan fitur matriks populasi piksel dan algoritma *K-Nearest Neighbor* sehingga mampu mengenali citra hasil foto walaupun menggunakan kamera yang kurang mumpuni.
3. Prediksi terhadap citra hasil deteksi foto dengan kualitas yang kurang baik berhasil dikenali dengan persentase 64,37575%
4. Prediksi terhadap citra yang memiliki resolusi terlalu tinggi dapat mengeluarkan hasil prediksi yang salah dikarenakan data yang dikandung sebuah piksel dalam citra dengan resolusi tinggi terlalu besar untuk dihitung dengan RGB yang kemungkinan mengeluarkan rata-rata *value* RGB 0,333% jika resolusi gambar terlalu tinggi.

### Daftar Pustaka

- [1] Candra Dewi , Dany Primanita Kartikasari , Yusi Tyroni Mursityo. 2014. Prediksi Cuaca Pada Data Time Series Menggunakan Adaptive Neuro Fuzzy Inference System (Anfis)., I(1), 18 – 24.
- [2] Krismiaji. 2010. Sistem Informasi Akuntansi. Yogyakarta : UPP AMP YKPN.
- [3] Kristanto Andri. 2008. Perancangan Sistem Informasi dan Aplikasinya. Yogyakarta: Gava Media.
- [4] Ricky Imanuel, Kusrini, M.Rudyanto Arief. 2014. Analisa Prediksi Tingkat Pengunduran Diri Mahasiswa Dengan Metode K-Nearest Neighbor.
- [5] Elsyantri Nana Suhendra, I Wayan Santiyasa, Agus Muliantara. 2015. Pengaruh Pca (Principle Component Analysis) Terhadap Klasifikasi Prakiraan Hujan Harian Di Daerah Kuta Selatan Menggunakan Algoritma Knn (K-Nearest Neighbor)., VIII(1), 8 – 14.